

**AMENDMENTS TO THE CLAIMS:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**LISTING OF CLAIMS**

1. (Currently Amended) A method for handling packet flows, comprising sequences of data packets, in a communication or computer system, the method comprising: assigning an exit number to each said packet; queuing said packets in buffer means; allocating said exit numbers to storage bins, each bin accepting a respective range of orders of exit numbers ~~(1, 3-2, 7-4)~~; ~~[[and]]~~ outputting the queued packets in a predetermined order according to an order list determined by said exit numbers assigned to said packets before said packets are queued; ~~the method characterised by the step,~~ and before said outputting step, of sorting the contents of a bin containing a first range of exit numbers ~~(3-2, 7-4)~~ into a plurality of bins each containing a smaller range of exit numbers ~~(3, 2, 7-6, 5-4)~~, the bins for higher order exit number packet records having a larger range than bins for lower order exit number packet records, and wherein said queuing step comprises placing packet records, each record containing information about its respective packet, together with their respective exit numbers in said bins.

2. (Original) A method as claimed in claim 1, wherein said sorting step is repeated until the contents of the bins are completely sorted.

3. (Previously Presented) A method as claimed in claim 1, wherein said queuing step comprises placing each said data packet together with its respective exit number in said buffer means, said buffer means comprising said storage bins.

4. (Cancelled)

5. (Currently Amended) A method as claimed in claim [[4]]1, wherein the packet records are of fixed length.
6. (Currently Amended) A method as claimed in claim [[4]]1, wherein the packet records are shorter than said packets.
7. (Cancelled)
8. (Currently Amended) A method as claimed in claim [[7]]1, wherein under circumstances in which a packet record is assigned an exit number ~~(2)~~ corresponding to a bin that is currently being emptied, that packet record is held in a specially created secondary bin ~~(2')~~ of the same exit number order for emptying after the said bin that is currently being emptied.
9. (Currently Amended) A method as claimed in claim [[4]]1, wherein the bins are FIFO buffers.
10. (Currently Amended) A method as claimed in claim [[4]]1, wherein the bins are LIFO stacks.
11. (Currently Amended) A method as claimed in claim [[4]]1, wherein the bins are a mixture of FIFO buffers and LIFO stacks.
12. (Original) A method as claimed in claim 1, wherein queue management is performed by (a) processing all of said bins in parallel and (b) inserting incoming data into a bin by means of a parallel processor.
13. (Original) A method as claimed in claim 12, wherein said parallel processor performing said inserting step (b) is an array processor.

14. (Original) A method as claimed in claim 13, wherein said array processor performing said inserting step (b) is a SIMD processor.

15. (Original) A method as claimed in claim 1, wherein said sorting step is carried out by a parallel processor.

16. (Original) A method as claimed in claim 15, wherein said parallel processor is an array processor.

17. (Original) A method as claimed in claim 16, wherein said array processor is a SIMD processor.

18. (Currently Amended) A data manager for handling packet flows, comprising sequences of data packets, in a communication or computer system, the system comprising: assigning means for assigning an exit number to each said packet; buffer means for queuing said packets; a series of storage bins; allocating means for allocating said exit numbers to said storage bins together with packet records, each record containing information about its respective packet, each bin accepting a respective range of orders of exit numbers (1, 3-2, 7-4), the bins for higher order exit number packet records having a larger range than bins for lower order exit number packet records; and output means for outputting the queued packets in a predetermined order according to an order list determined by said exit numbers assigned to said packets before said packets are queued; the system characterised by sorting means upstream of said output means for sorting the contents of a bin containing a first range of exit numbers (3-2, 7-4) into a plurality of bins each containing a smaller range of exit numbers (3, 2, 7-6, 5-4).

19. (Original) A data manager as claimed in claim 18, wherein said allocating means comprises a parallel processor.

20. (Original) A data manager as claimed in claim 19, wherein said sorting means comprises a parallel processor.

21. (Previously Presented) A data manager as claimed in claim 19, wherein said parallel processor is an array processor.

22. (Original) A data manager as claimed in claim 21, wherein said parallel processor is a SIMD processor.

23. (Previously Presented) A data manager as claimed in claim 18, wherein said buffer means is adapted to queue said data packets together with their respective exit numbers.

24. (Previously Presented) A data manager as claimed in claim 18, wherein said buffer means comprises said series of bins, and wherein said bins are adapted to receive packet records, each record containing information about a respective packet, together with the respective exit number, and said buffer means is adapted to queue said packets.

25. (Original) A data manager as claimed in claim 18, wherein said sorting means is adapted to repeat sorting said bins until the contents of the bins are completely sorted.

26. (New) A method for handling packet flows, comprising sequences of data packets, in a communication or computer system, the method comprising:

assigning an exit number to each said packet;

queuing said packets in a buffer;

allocating said exit numbers to storage bins, each bin accepting a respective range of orders of exit numbers;

outputting the queued packets in a predetermined order according to an order list determined by said exit numbers assigned to said packets before said packets are queued; and,

before said outputting step, sorting the contents of a bin containing a first range of exit numbers into a plurality of bins each containing a smaller range of exit numbers,

wherein said queuing step comprises placing packet records, each record containing information about its respective packet, together with their respective exit numbers in said bins; and

wherein the bins are a mixture of FIFO buffers and LIFO stacks.